

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-297702

(43)Date of publication of application : 18.11.1997

(51)Int.Cl.

G06F 12/00

G06F 13/00

(21)Application number : 08-110811

(71)Applicant : CANON INC

(22)Date of filing : 01.05.1996

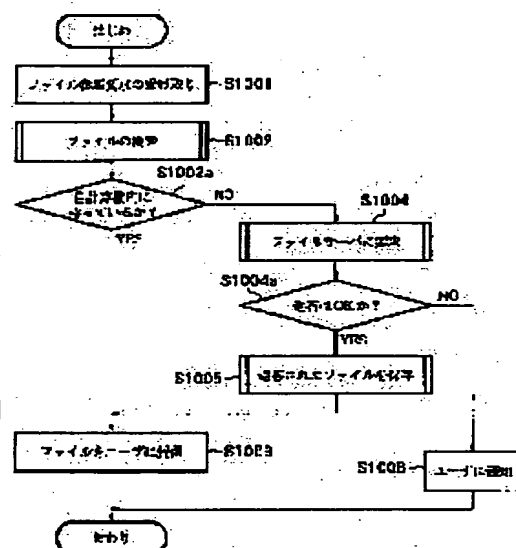
(72)Inventor : KUROSAWA TAKAHIRO

(54) PROCESSOR AND SYSTEM FOR INFORMATION PROCESSING AND THEIR CONTROL METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To provide the information processing system which is superior in durability to failure by copying necessary files present on a file server to a storage device inside its own device.

SOLUTION: A client-side computer (client computer) connected to a file server in a communicable state through a computer network retrieves a file which is requested to be referred to in the client computer (Steps S1001 and S1002). Then when the file is not found, the client computer requests the file of the file server through the computer network (step S1004). When a file sent from the file server at the request is received through the computer network, this file is saved and then provided to a user and specific processes are carried out (steps S1005 and S1003).



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-297702

(43)公開日 平成9年(1997)11月18日

(51)IntCl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/00	5 3 3		G 0 6 F 12/00	5 3 3 J
13/00	3 5 1		13/00	3 5 1 E

審査請求 未請求 請求項の数19 O L (全 16 頁)

(21)出願番号 特願平8-110811

(22)出願日 平成8年(1996)5月1日

(71)出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72)発明者 黒澤 貴弘

東京都大田区下丸子3丁目30番2号 キヤ

ノン株式会社内

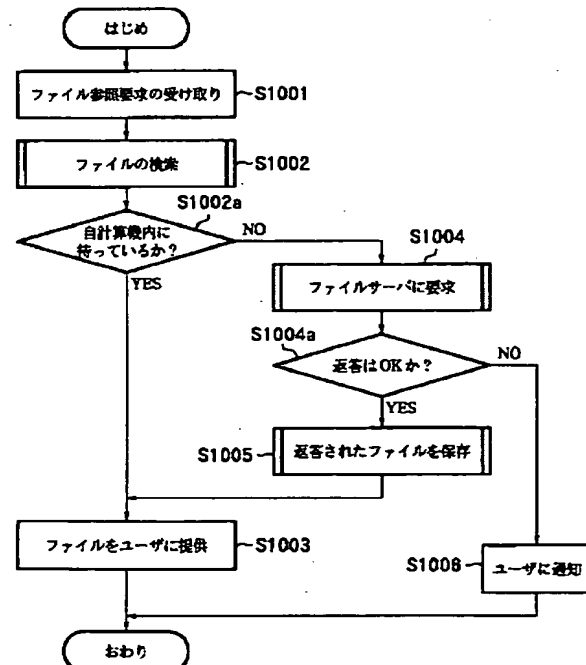
(74)代理人 弁理士 大塚 康徳 (外1名)

(54)【発明の名称】 情報処理装置及びシステム及びその制御方法

(57)【要約】

【課題】ファイルサーバ上に存在する必要なファイルを、自装置内の記憶装置に複製することを可能とし、耐故障性に優れる情報処理システムを提供する。

【解決手段】計算機ネットワークを介してファイルサーバと通信可能に接続されたクライアント側の計算機（クライアント計算機）において、ファイルの参照が要求されると参照が要求されたファイルをクライアント計算機内で検索する（ステップS1001、S1002）。そして、該当するファイルが検索されなかった場合は、ファイルサーバに対して計算機ネットワークを介して該ファイルを要求する（ステップS1004）。そして、当該要求の応答としてファイルサーバより送信されたファイルを計算機ネットワークを介して受信すると、この受信したファイルを保存した後に、該受信したファイルをユーザに提供して所定の処理を遂行する（ステップS1005、S1003）。



1

【特許請求の範囲】

【請求項 1】 通信手段を介して外部装置と通信が可能な情報処理装置であって、

参照が要求されたファイルを当該装置内で検索する検索手段と、

前記検索手段によって前記ファイルが検索されなかった場合、外部装置に対して前記通信手段を介して該ファイルを要求する要求手段と、

前記要求手段による要求の応答として送信されたファイルを前記通信手段を介して受信し、この受信したファイルを保存する保存手段と、

前記受信したファイルを参照して所定の処理を遂行する制御手段とを備えることを特徴とする情報処理装置。

【請求項 2】 前記検索手段は、参照が要求されたファイルが検索された場合に、前記外部装置との間で該ファイルの一貫性を確認し、

前記要求手段は、前記検索手段によって前記ファイルが検索されなかった場合、或は前記検索手段によって検索されたファイルの一貫性が保たれていない場合に、前記外部装置に対して前記通信手段を介して当該ファイルを要求することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】 前記要求手段は、前記ファイルをそのファイル名を用いて要求することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 4】 前記保存手段は、前記受信したファイルを二次記憶装置へ保存することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 5】 前記保存手段は、前記受信したファイルを主メモリの一部に保存することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 6】 前記受信したファイルを前記保存手段によって保存する際に、保存先の記憶媒体に十分な空き容量があるか否かを判定する判定手段と、

前記判定手段で十分な空き容量が存在しないと判定された場合に、前記記憶体上の 1 つ又は複数のファイルを削除する削除手段とを更に備えることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 7】 前記削除手段は、前記記憶媒体に記憶された各ファイルの参照頻度を示す統計情報に基づいて、参照頻度が所定値に満たないファイルを削除することを特徴とする請求項 6 に記載の情報処理装置。

【請求項 8】 前記統計情報は、各作業の種類毎に、長期的な参照回数を蓄積したものであることを特徴とする請求項 7 に記載の情報処理装置。

【請求項 9】 前記記憶媒体に記憶された各ファイルについて、夫々の参照回数に基づいて前記統計情報を更新する更新手段を更に備えることを特徴とする請求項 7 に記載の情報処理装置。

【請求項 10】 少なくとも 1 つのサーバ装置とクライ

2

アント装置が通信手段を介して通信可能に接続された情報処理システムであって、

クライアント装置において参照が要求されたファイルを当該クライアント装置内で検索する検索手段と、

前記検索手段によって前記ファイルが検索されなかった場合、前記サーバ装置に対して前記通信手段を介して該ファイルを要求する要求手段と、

前記サーバ装置において、前記要求手段によって要求されたファイルを獲得し、この獲得したファイルを前記通信手段を介して前記クライアント装置へ送信する送信手段と、

前記クライアント装置において、前記送信手段で送信されたファイルを受信し、この受信したファイルを保存する保存手段と、

前記クライアント装置において、前記受信したファイルを参照して所定の処理を遂行する制御手段とを備えることを特徴とする情報処理システム。

【請求項 11】 前記サーバ装置における前記送信手段は、汎用データベースシステムを利用して実行されることを特徴とする請求項 10 に記載の情報処理システム。

【請求項 12】 前記サーバ装置における前記送信手段は、汎用のネットワークファイルシステムを利用して実行されることを特徴とする請求項 10 に記載の情報処理システム。

【請求項 13】 通信手段を介して外部装置と通信が可能な情報処理装置の制御方法であって、

参照が要求されたファイルを当該装置内で検索する検索工程と、

前記検索工程によって前記ファイルが検索されなかった場合、外部装置に対して前記通信手段を介して該ファイルを要求する要求工程と、

前記要求工程による要求の応答として送信されたファイルを前記通信手段を介して受信し、この受信したファイルを保存する保存工程と、

前記受信したファイルを参照して所定の処理を遂行する制御工程とを備えることを特徴とする情報処理装置の制御方法。

【請求項 14】 前記受信したファイルを前記保存工程によって保存する際に、保存先の記憶媒体に十分な空き容量があるか否かを判定する判定工程と、

前記判定工程で十分な空き容量が存在しないと判定された場合に、前記記憶体上の 1 つ又は複数のファイルを削除する削除工程とを更に備えることを特徴とする請求項 13 に記載の情報処理装置の制御方法。

【請求項 15】 前記削除工程は、前記記憶媒体に記憶された各ファイルの参照頻度を示す統計情報に基づいて、参照頻度が所定値に満たないファイルを削除することを特徴とする請求項 14 に記載の情報処理装置の制御方法。

【請求項 16】 前記統計情報は、各作業の種類毎に、

3

長期的な参照回数を蓄積したものであることを特徴とする請求項15に記載の情報処理装置の制御方法。

【請求項17】 前記憶媒体に記憶された各ファイルについて、夫々の参照回数に基づいて前記統計情報を更新する更新工程を更に備えることを特徴とする請求項15に記載の情報処理装置の制御方法。

【請求項18】 少なくとも1つのサーバ装置とクライアント装置が通信手段を介して通信可能に接続された情報処理システムの制御方法であって、

クライアント装置において参照が要求されたファイルを当該クライアント装置内で検索する検索工程と、

前記検索工程によって前記ファイルが検索されなかった場合、前記サーバ装置に対して前記通信手段を介して該ファイルを要求する要求工程と、

前記サーバ装置において、前記要求工程によって要求されたファイルを獲得し、この獲得したファイルを前記通信手段を介して前記クライアント装置へ送信する送信工程と、

前記クライアント装置において、前記送信工程で送信されたファイルを受信し、この受信したファイルを保存する保存工程と、

前記クライアント装置において、前記受信したファイルを参照して所定の処理を遂行する制御工程とを備えることを特徴とする情報処理システムの制御方法。

【請求項19】 通信手段を介して外部装置と通信が可能な情報処理装置においてファイル共有を実現するための制御プログラムが格納されたコンピュータ可読メモリであって、

参照が要求されたファイルを当該装置内で検索する検索工程のコードと、

前記検索工程によって前記ファイルが検索されなかった場合、外部装置に対して前記通信手段を介して該ファイルを要求する要求工程のコードと、

前記要求工程による要求の応答として送信されたファイルを前記通信手段を介して受信し、この受信したファイルを保存する保存工程のコードと、

前記受信したファイルを参照して所定の処理を遂行する制御工程のコードとを備えることを特徴とするコンピュータ可読メモリ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、複数の情報処理装置間においてファイルを共有する情報処理装置及びシステム及びその制御方法に関する。

【0002】

【従来の技術】 近年、高速なネットワークで結合された高性能計算機群で構成されたコンピュータシステムが普及してきた。このような状況において、同様なファイル構成をもつ計算機間でファイルを共有することが行われている。

4

【0003】 オペレーティングシステムなどを構成するファイル群の典型的な保持方法は、各計算機の二次記憶装置(各計算機のハードディスク等)に、予め決められた構成で蓄積しておく方法である。たとえば、UNIXなどのオペレーティングシステムを実装したコマンドのファイル群は、標準的なシステム構成で、50~150メガバイト程の二次記憶装置の領域を必要としていた。通常の利用において、これらオペレーティングシステムを構成するファイル群は、そのファイル群全体がローカルなファイルシステムに保持されている。

【0004】 しかし、ネットワークを介して複数の計算機が結ばれた環境では、NFS(ネットワークファイルシステム)などにより一部のファイル群を共有することも行なわれていた。このような共有では、実行効率の面から、ファイルシステム(ボリューム)を単位とした共有形態が多かった。例えば、Sun Microsystemsが提案したNFS(ネットワークファイルシステム)や、CMUのAFS(アンドリュウファイルシステム)などである。

【0005】 さらに、主メモリ装置の容量に余裕のある計算機では、RAMディスクなどの技法が使われ、主メモリの一部を二次記憶装置のように利用する事で、二次記憶装置へのアクセス頻度を低減する事が行なわれていた。例えば、SunOSにおけるtmpfsである。

【0006】

【発明が解決しようとしている課題】 ファイルシステム(ボリューム)を単位とした共有を行なうNFSなどのファイル共有方法では、ファイルサーバが利用できない場合にクライアントも動作できなくなるという問題、すなわち耐故障性の問題がある。つまり、ファイルシステムを単位とした共有方法では、実行効率を向上させるために、ファイルシステム内の内部的な参照方法(たとえば、v-node)を利用するため、ファイルサーバの介在無しには、クライアント側でファイルの参照が困難になってしまう。

【0007】 さらに、クライアント側計算機の記憶容量が限られる状況において使われるスワップアウト時のファイル選択処理がLRUなどに代表される短期的な統計情報を用いて行われると、将来に渡って高い頻度で利用されるファイルをもスワップアウトしてしまうという問題がある。

【0008】 本発明は上記の問題に鑑みてなされたものであり、ファイルサーバ上に存在する必要なファイルを、自装置内の記憶装置に複製することを可能とし、耐故障性に優れる情報処理システム及び装置及びその方法を提供することを目的とする。

【0009】 また、本発明の他の目的は、ファイルのスワップアウト時において、スワップアウトすべきファイルを適切に選択することが可能な情報処理システム及び装置及びその制御方法を提供することにある。

5

【0010】

【課題を解決するための手段】上記の目的を達成するための本発明の情報処理装置は以下の構成を備えている。即ち、通信手段を介して外部装置と通信が可能な情報処理装置であって、参照が要求されたファイルを当該装置内で検索する検索手段と、前記検索手段によって前記ファイルが検索されなかった場合、外部装置に対して前記通信手段を介して該ファイルを要求する要求手段と、前記要求手段による要求の応答として送信されたファイルを前記通信手段を介して受信し、この受信したファイルを保存する保存手段と、前記受信したファイルを参照して所定の処理を遂行する制御手段とを備える。

【0011】また、好ましくは、前記検索手段は、参照が要求されたファイルが検索された場合に、前記外部装置との間で該ファイルの一貫性を確認し、前記要求手段は、前記検索手段によって前記ファイルが検索されなかった場合、或は前記検索手段によって検索されたファイルの一貫性が保たれていない場合に、前記外部装置に対して前記通信手段を介して当該ファイルを要求する。外部装置との間で共有されているファイルの一貫性が保た

れるからである。

【0012】また、好ましくは、前記要求手段は、前記

ファイルをそのファイル名を用いて要求する。

【0013】また、好ましくは、前記保存手段は、前記

受信したファイルを二次記憶装置へ保存する。

【0014】また、好ましくは、前記保存手段は、前記

受信したファイルを主メモリの一部に保存する。

【0015】また、好ましくは、前記受信したファイルを前記保存手段によって保存する際に、保存先の記憶媒体に十分な空き容量があるか否かを判定する判定手段と、前記判定手段で十分な空き容量が存在しないと判定された場合に、前記記憶媒体上の1つ又は複数のファイルを削除する削除手段とを更に備える。記憶領域に制限がある場合でも、参照要求の発生したファイルを保存するための領域を生成し、保存することが可能となるからである。

【0016】また、好ましくは、前記削除手段は、前記記憶媒体に記憶された各ファイルの参照頻度を示す統計情報に基づいて、参照頻度が所定値に満たないファイルを削除する。参照頻度の小さいファイルを削除することにより、有用なファイルを残すことが可能となり、記憶媒体の有効利用が図られる。

【0017】また、好ましくは、前記統計情報は、各作業の種類毎に、長期的な参照回数を蓄積したものである。ファイルの参照頻度情報を各作業毎に保持するので、記憶媒体に保存されるファイルの構成が各作業に適したものとなり、記憶媒体の使用効率が更に向上する。

【0018】また、好ましくは、前記記憶媒体に記憶された各ファイルについて、夫々の参照回数に基づいて前記統計情報を更新する更新手段を更に備える。統計情報

6

が、各ユーザの使用状況により適応したものとなるからである。

【0019】以上のような削除手段、統計情報により、例えばファイルのスワップアウト時において、スワップアウトすべきファイルを適切に選択することが可能となる。

【0020】また、上記の目的を達成するための本発明の情報処理システムは、少なくとも1つのサーバ装置とクライアント装置が通信手段を介して通信可能に接続された情報処理システムであって、クライアント装置において参照が要求されたファイルを当該クライアント装置内で検索する検索手段と、前記検索手段によって前記ファイルが検索されなかった場合、前記サーバ装置に対して前記通信手段を介して該ファイルを要求する要求手段と、前記サーバ装置において、前記要求手段によって要求されたファイルを獲得し、この獲得したファイルを前記通信手段を介して前記クライアント装置へ送信する送信手段と、前記クライアント装置において、前記送信手段で送信されたファイルを受信し、この受信したファイルを保存する保存手段と、前記クライアント装置において、前記受信したファイルを参照して所定の処理を遂行する制御手段とを備える。

【0021】また、上記の目的を達成するための本発明による情報処理装置の制御方法は以下の工程を備える。

即ち、通信手段を介して外部装置と通信が可能な情報処理装置の制御方法であって、参照が要求されたファイルを当該装置内で検索する検索工程と、前記検索工程によって前記ファイルが検索されなかった場合、外部装置に対して前記通信手段を介して該ファイルを要求する要求工程と、前記要求工程による要求の応答として送信されたファイルを前記通信手段を介して受信し、この受信したファイルを保存する保存工程と、前記受信したファイルを参照して所定の処理を遂行する制御工程とを備える。

また、上記の目的を達成するための本発明による情報処理システムの制御方法は以下の工程を備える。即ち、少なくとも1つのサーバ装置とクライアント装置が通信手段を介して通信可能に接続された情報処理システムの制御方法であって、クライアント装置において参照が要求されたファイルを当該クライアント装置内で検索する検索工程と、前記検索工程によって前記ファイルが検索されなかった場合、前記サーバ装置に対して前記通信手段を介して該ファイルを要求する要求工程と、前記サーバ装置において、前記要求工程によって要求されたファイルを獲得し、この獲得したファイルを前記通信手段を介して前記クライアント装置へ送信する送信工程と、前記クライアント装置において、前記送信工程で送信されたファイルを受信し、この受信したファイルを保存する保存工程と、前記クライアント装置において、前記受信したファイルを参照して所定の処理を遂行する制御工程とを備える。

【0022】

【発明の実施の形態】以下、添付の図面を参照して本発明の好適な一実施形態を説明する。

【0023】＜第1の実施形態＞第1の実施形態では、後述のファイル参照複製プログラムがオペレーティングシステムの一部として実装されたクライアント側の計算機（以下、クライアント計算機という）による、ファイルの獲得と保存の処理について説明する。クライアント計算機は、ファイル参照複製プログラムを実行することにより、自らは装備していないコマンドを利用しようとした際に、そのコマンドを実装したファイルをファイルサーバから取り寄せ、当該クライアント計算機内に複製し、しかる後に当該ファイルを利用する（即ち、当該コマンドを実行する）。なお、本第1の実施形態では、ファイルは、UNIXにおけるパス名のように名前によって識別されるものとする。

【0024】以下、まず本実施形態を説明するための装置の構成について説明し、次に、その動作手順について述べる。

【0025】〔1〕構成図の説明

図1は、第1の実施形態による計算機システム及び計算機の構成を説明するブロック図である。同図において100は、計算機ネットワークに接続された計算機を示す。ここで、計算機100は、以下の構成を備える。

【0026】101は入力装置であり、本実施形態上で動作するプログラムの利用者が、コマンドやデータを入力する。入力装置101は、キーボード、マウス、パッド、あるいは、マイクロフォンなどを備える。102はCPUであり、本実施形態で説明する処理を含む、計算機100における各種処理を実現する。なお、ここでは1つのCPUを使っているが、それぞれのプログラムの実行に独立なプロセッサを利用するようなハードウェア形態であっても構わない。103は、本実施形態における処理の内容をユーザに表示するための出力装置であり、CRTディスプレイあるいはプロジェクターなどで構成される。ここでは、説明を簡単にするため、単一の表示装置を使っているが、それぞれに独立な表示装置を使ってもかまわない。

【0027】104は記憶装置であり、本実施形態で説明する処理を実現するための制御プログラム、ならびに、それらの制御プログラムが利用するデータを保存する。ここでは、説明を簡単にするために単一の記憶装置を使っているが、記憶対象毎に異なった記憶装置を用いるようにしてもよいし、ハードディスクなどの二次記憶装置を含む階層化された記憶装置であってもよい。また、この記憶装置104は、一時的に使われるデータを記憶する主記憶装置と大量のデータを永続的に保持しておく二次記憶装置とに分かれていてもよい。更に、記憶装置104の状態に関係なく永続的なデータを保存するROMなどの読み出し専用メモリが記憶装置104に含

まれていてもよい。

【0028】105は通信装置であり、計算機ネットワークに接続するFDDIコントローラなどである。106は計算機バスであり、以上に述べた構成要素100～105ならびに記憶媒体読み取り装置107を結合する計算機バスである。

【0029】計算機200、300は、上述の計算機100と同様の構成を備えた計算機である。ここでは、3つの計算機を利用する構成について説明するが、システムに存在する計算機の数は、2つあるいは4つ以上であってもよい。

【0030】5000は、本実施形態で説明する複数の計算機を接続するための計算機ネットワークであり、例えば、EthernetやFDDIなどである。本実施形態では、汎用の計算機ネットワークを利用した実現形態について説明しているため、その通信形態として計算機ネットワークを用いているが、これは、ISDNなどの広域回線であってもよい。

【0031】図2はクライアント計算機の記憶装置104に格納される主な制御プログラムを示す図である。記憶装置104に保持される制御プログラムは、計算機のオペレーティングシステム10、ファイル参照複製プログラム12、ファイル検索プログラム20、ファイル参照プログラム21、ファイル保存プログラム22などの制御プログラムである。また、ファイル参照頻度の統計情報40も記憶装置104に保持されている。ここで、統計情報は、計算機の1回の使用期間中における参照頻度を示すものではなく、ファイル参照複製プログラム12をインストールした後の継続的、かつ長期的な参照頻度を示すものである。

【0032】なお、これらの制御プログラムならびにデータは、フロッピーディスク装置あるいはCD-ROM装置などの外部記憶媒体読み取り装置107から読み込まれてもよい。

【0033】また、図3はファイルサーバ側の計算機（以下、サーバ計算機という）の記憶装置104に格納される主な制御プログラムを示す図である。サーバ計算機の記憶装置104には、オペレーティングシステム10に加えて、ファイルサーバプログラム11が格納される。

【0034】〔2〕動作手順の概要

以上の各制御プログラムの関連を説明すると次の通りである。クライアント計算機において、コマンド等の入力によってファイル参照要求が発生すると、オペレーティングシステム10によってファイル参照複製プログラム12が起動される。ファイル参照複製プログラム12は、ファイル参照要求によってファイル検索プログラム20を起動し、自計算機内に要求されたファイルが格納されているかを調べる。該当ファイルが自計算機に格納されていれば、そのファイルをオペレーティングシステ

9

ム10に通知する。

【0035】一方、該当ファイルが自計算機内に格納されていなければ、ファイル参照複製プログラム12はファイル参照プログラム21を起動し、サーバ計算機にファイルの参照要求を行う。サーバ計算機のファイルサーバプログラム11は、クライアント計算機からのファイル参照要求により、該当するファイルを検索し、これをクライアント計算機へ転送する。

【0036】クライアント計算機では、サーバ計算機より該当するファイルが送信されると、ファイル保存プログラム22を起動して、当該ファイルを保存した後、当該ファイルを用いて入力されたコマンドを実行する。

【0037】次に、本実施形態における装置の動作手順について、フローチャートを用いて更に詳細に説明する。

【0038】図4は本実施形態におけるファイル参照複製プログラム12の制御手順を表すフローチャートである。上述のように、ファイル参照複製プログラム12は、クライアント計算機（例えば計算機100）のオペレーティングシステム10の一部として組み込まれている。この制御プログラムは、ユーザからオペレーティングシステム10にファイル参照要求が出された際にオペレーティングシステム10から呼び出され、上記のシステム構成上で動作する。

【0039】まず、ユーザから発行されたファイル参照要求を受けとる（ステップS1001）。このファイル参照要求が発生するのは、例えば、アプリケーションの起動時に、当該起動コマンドを実装したプログラムが入っているファイルを要求する場合などである。次に、当該ファイル参照要求に応じて、ファイル検索プログラム20を起動し、自計算機内において与えられた名前のファイルを探す（ステップS1002）。ここで、もし、要求されたファイルが既に自計算機内に存在するのであれば、ステップS1002aからステップS1003へ進み、当該ファイルをユーザに（オペレーティングシステム10に）提供する。

【0040】反対に、当該ファイルを自計算機内に持っていない場合は、ステップS1002aからステップS1004へ進み、ファイル参照プログラム21の起動により、ファイルサーバに該当ファイルの提供を要求する。そして、ファイルサーバから該当するファイルを提供できる旨の通知（OKの通知）を受けたなら、ファイルサーバより転送されたファイルをファイル保存プログラム22により自計算機内に保存する。その後、当該ファイルはユーザに提供される（ステップS1004a、S1005）。一方、ファイルサーバにも該当するファイルが無ければ、ファイルサーバよりの返答はNGとなるので、ユーザにその旨を通知する（ステップS1004a、S1006）。

【0041】本第1の実施形態の特徴は、NFSなどの

10

ようにファイルシステム（ボリューム）を単位としてファイルを参照するのとは異なり、ファイル保存プログラム22を使って、個々のファイルを陽に自らの計算機内に固定する点にある。つまり、ファイルサーバ上に構成されたファイル群の一部分を、ファイルを単位として、必要に応じてクライアント計算機上に複製することで、クライアント側のファイル群の内容を成長させる。例えば図9はファイルサーバ側のファイル構成例を説明する図であり、図10はクライアント側のファイル構成例を説明する図である。図9のようなファイル群がサーバ計算機に保持されている状況で、クライアント計算機の要求に応じて、その一部分がファイルを単位としてクライアント計算機に複製される。その結果、図10に示すようなファイル群がクライアント計算機に形成され、保持される。

【0042】なお、ここでは、説明を簡単にするため、クライアント計算機のオペレーティングシステムが、最初にファイルサーバと通信する際のファイルサーバの同定方法について言及していない。しかしながら、これは、UNIXのNIS（Network Information Service）などで採用されているのと同様の手法で実現できる。すなわち、クライアント計算機が、機能提供可能なサーバを探索するためのメッセージをブロードキャストし、その機能提供が可能なファイルサーバがそれに答えるようにプログラムすることで実現できる。

【0043】次に、ファイルサーバ側計算機で動作するファイルサーバプログラム11の動作概要について説明する。図5は本実施形態におけるファイルサーバプログラムの制御手順を説明するフローチャートである。このプログラムは、ネットワーク内の少なくとも一つ以上の計算機（たとえば、計算機300）で動作してサービスを提供している必要がある。

【0044】まず、クライアント計算機（例えば、計算機100）からのファイル参照要求を受け取る（ステップS2001）。ここで、ファイルサーバプログラム11が受け取るファイル参照要求は、ファイル参照プログラム21によって発行されたものである。次に、当該サーバ計算機内において、ファイル参照要求によって与えられた名前のファイルを検索する（ステップS2002）。ここで、もし、当該計算機が目的のファイルをもっているなら、そのファイルを要求しているクライアントに送り返す（ステップS2002a、S2003）。一方、該当するファイルが存在しなければ、当該参照要求を発行したクライアントにその旨を通知する（ステップS2002a、S2004）。

【0045】なお、ここでは専用のファイルサーバプログラム11が動作している例について説明しているが、汎用のデータベースシステムを用いて、個々のファイルを提供する方法も考えられる。また、クライアント側のプログラムを、ftp（FileTransfer Program）などの

ネットワークファイル転送プロトコルを利用するように変更する事で、ファイルサーバプログラムにftpサーバなどの汎用のネットワークファイルサーバを利用することも容易に実現できる。

【0046】以下では、上記の動作手順の概要において用いた各処理（ファイル検索プログラム20、ファイル参照プログラム21、ファイル保存プログラム22）の動作を、フローチャートを用いながら説明する。

【0047】[3]ファイル検索プログラム20

ここでは、計算機内のファイルを探すためのファイル検索処理を実現するプログラムについて説明する。図6は本実施形態におけるファイル検索プログラムの制御手順を説明するフローチャートである。本実施形態では、ファイルは、UNIXにおけるパス名のように名前によって識別されるものとする。

【0048】まず、自計算機内に保持している全てのファイルの名前と、ファイル参照要求によって与えられたファイルの名前との比較を行なう（ステップS3001）。次に、該当するファイルが見つかったら、ファイルサーバに問合せ、そのファイルの一貫性に関する妥当性を検査する（ステップS3001a、S3002）。この際、ファイルサーバにアクセスできないならば、ファイルサーバにトラブルがあったものと判断し、そのファイルを検索結果として回答する（ステップS3002a、S3003）。

【0049】また、ファイルサーバへのアクセスが可能であり、ファイルサーバからの返答が『一貫性に関して妥当である』なら、そのファイルを検索結果として回答する（ステップS3002a、S3002b、S3003）。この場合、当該ファイルを使用すれば良いので、図4のステップS1002aの分岐処理によって、ステップS1003へ進む。一方、該当するファイルが見つからなかったり、『一貫性に関して妥当でない』との返答なら、その旨を回答する（ステップS3002b、S3004）。この場合（一貫性に関して妥当でない場合）、ファイルサーバから当該ファイルを獲得する必要があるため、図4のステップS1002aの分岐処理によって、ステップS1004へ進むことになる。

【0050】なお、ここで、サーバ計算機とクライアント計算機による一貫性確認は、例えば次のように行われる。即ち、クライアント計算機は、一貫性を確認すべきファイルのファイル名と、当該ファイルの作成日時をサーバ計算機へ通知する。サーバ計算機は、通知されたファイル名を有するファイルを検索し、その作成日時を獲得し、クライアント計算機より通知された作成日時と比較する。そして、この比較の結果、両作成日時が一致すれば『一貫性に関して妥当である』旨を、作成日時が一致しなければ『一貫性に関して妥当でない』旨をクライアント装置へ通知する。

【0051】ここでは、ファイル検索時に、そのファイ

ルの妥当性を検査して一貫性を保持する例について説明しているが、ファイルサーバが、ファイルの変更を検出した時点で、そのファイルの複製を保持している全てのクライアントに変更通知を行い、その変更通知に応じて各クライアント自身がファイルを無効化（invalidate）することで、一貫性を保持する方法も容易に実現できる。

【0052】また、ここでは、全てのファイルに関して名前による比較を行なっているが、それ以外の識別方法を用いるようにしても良い。たとえば、オブジェクト指向データベースシステムにおけるオブジェクト識別子（object identifier）などが挙げられる。また、適切なインデックスを作成する事により、探索空間を小さくする事も考えられる。

【0053】[4]ファイル参照プログラム21

次に、自計算機内にファイル参照要求で要求されたファイルが存在しない場合に、ファイルサーバに対して条件を満たすファイルの提供を要求するファイル参照プログラムについて説明する。図7はファイル参照プログラムの制御手順を説明するフローチャートである。

【0054】まず、求めるファイルを検し出す識別条件となる情報をファイルサーバに送る（ステップS4001）。本実施形態では、ファイルの名前により識別するので、求めるファイルの名前をファイルサーバに渡すことになる。次に、ファイルサーバからの返答を待つ（ステップS4002）。

【0055】ここでは、ファイル名を識別条件として指定しているが、他の識別条件を指定するようにしてもよいことは上述した通りである。即ち、たとえば、オブジェクト指向データベースシステムにおけるオブジェクト識別子（object identifier）などが挙げられる。

【0056】[5]ファイル保存プログラム22

ここでは、ファイル単位の利用統計情報を使ったスワップアウト処理を含むファイル保存プログラムについて説明する。図8は本実施形態におけるファイル保存プログラムの制御手順を説明するフローチャートである。なお、ここでは、ファイル保存用に割り当てられた記憶装置として、記憶装置104の二次記憶装置の一部を割り当てているものとする。また、ここでは、ファイル参照頻度の統計情報40がクライアント計算機に固定されて存在する場合について説明している。しかしながら、ファイルサーバが常にファイル参照を監視して利用統計情報を収集している場合には、そのファイルサーバが保持している動的な利用統計情報をも加味してもよい。即ち、記憶装置104に保持された静的な統計情報に、ファイルサーバが保持している動的な利用統計情報を加味して、各ファイルのランク付けを逐次更新させてもよい。また、上記統計情報は、クライアント計算機において、各ファイルに対する実際の参照回数に従って更新されてもよい。

13

【0057】まず、ファイルサーバより提供されたファイルのサイズを求める(ステップS5001)。次に、このファイルサイズに基づいて、ファイル保存用に割り当てられた記憶装置104の記憶領域に余裕があるかどうか判定する(ステップS5002)。ここでは、ファイルサイズ以上の空き領域があれば、余裕があると判定する。判定の結果、領域に余裕があるなら、受けとったファイルを、ファイルサーバ上と同様の構成(たとえば、同じディレクトリ構成)で記憶装置104に記録する(ステップS5002a、S5003)。

【0058】一方、領域に余裕がないと判定された場合は、ファイル参照頻度の統計情報40を参照して、現在利用されていないファイルの内、参照頻度ランクが所定ランクより低いファイルを消去する。これにより、空き領域の確保を試みる(ステップS5002a、S5004)。この試みの結果、もし、空き領域が確保できるなら、当該ファイルをそこに記憶する(ステップS5004a、S5003)。スワップアウトを実行しても十分な記憶領域を確保できない場合は、ファイルの保存に失敗した旨を利用者に通知する(ステップS5004a、S5006)。

【0059】なお、本実施形態では、説明を簡単にするために、ファイル参照頻度の統計情報40は、全てのクライアントに予め保持されているものとする。保持されている統計情報40は、例えば、『プログラム開発』や『文書作成』というような作業内容毎にファイル参照頻度の統計情報が格納される。ユーザは、各作業の開始に先立って、これから行う作業の種別(プログラム開発、文書作成等)を入力する。クライアント装置では指定された作業種別に対応する統計情報を用いて上述の処理を行う。なお、クライアント装置に指定された作業種別に対応する統計情報がない場合は、指定された種別に対応する統計情報をサーバ装置より獲得する。

【0060】例えば、図11A、11Bは『プログラム開発』という作業内容を対象としたファイル参照頻度の統計情報の一例を示す図である。図中、行単位を見ると、参照頻度ランクを基準として、いくつかのファイル群にグルーピングされている。左端の列に示した数字が参照頻度ランクを表し、大きいほど、参照頻度が高い事を示す。中央の列が、参照頻度を表す係数であり、そして、右端の列が、ファイル名を示している。なお、参照頻度に対する参照頻度ランクは、対数によって設定されているものとしている。従って、上述のスワップアウトでは、参照頻度ランクが1のファイルを削除するようにする。

【0061】このファイル保存プログラム22では、クライアント計算機に保持されるファイルを必要なファイルのみに極小化し、以降の参照を最適化する事で高速な記憶装置の有効利用を図っている。また、保持するファイルは、クライアント計算機に保持されたファイル単位

14

の利用統計情報に依存して、利用率が高いと予測されるファイルを選択している。別の見方をすれば、利用率の低いファイルは削除対象となる。ここで、この利用統計情報は、長期的統計の蓄積から、システムの作業内容毎に設定されたファイル参照頻度に基づく点が本実施形態の特徴である。従って、この点において、本実施形態の利用統計情報は、仮想記憶管理方式などに利用されるLRUのような短期的な参照履歴を用いた処理とは異なる。

10 【0062】なお、本実施形態では、ファイルサイズ以上の空き領域があれば領域に余裕があると判定したが、他の判定基準も考えられる。例えば、UNIXにおけるファイルシステム管理のように、スーパーユーザ以外の操作では、領域全体の90%までの利用しか許さないなどの基準が考えられる。このように、本実施形態のファイルサーバとして、実行クライアント制限等のようなライセンス管理を行うようにしても良い。

【0063】ここでは、ファイルサーバ上と同様の構成(たとえば、ディレクトリ構成)で、個々のファイルを保存する例について述べたが、クライアント計算機においてサーバ計算機とは異なった構成によってファイルを保存する事も考えられる。例えば、サーバ計算機側のファイル構成がUNIXのファイルシステムのように多段階の階層構造を持つ場合でも、クライアント計算機側では、ハッシュ表などを用いたフラットな構成で保持する事が考えられる。例えば、ディレクトリ階層の文字列表現(いわゆる、パス名)を、文字列ハッシュ関数を用いてハッシュし、ハッシュ表へのエントリを計算する。そして、ハッシュ表には、エントリ番号とオリジナルのパス名とファイル実体へのポインタを保持する。なお、文字列ハッシュ関数としては、文字列を4バイト毎に区切り、それらをExclusive-ORで、integerに変換する方法を採る。このように、本実施形態においては、ファイルサーバにおける編成とは異なった編成で複製ファイルを保存するようにしても良い。

【0064】上記実施形態では、オペレーティングシステムが提供する標準的なコマンド群のファイル利用について説明したが、これは、新たにインストールしようとするソフトウェア製品についても適用できる。すなわち、従来では、各クライアント計算機に、そのソフトウェア製品を個別にインストールする必要があったが、本発明の方法を利用すれば、ファイルサーバにのみソフトウェア製品をインストールすれば良く、クライアント計算機では、それを利用するように要求を出すだけで、その複製がクライアント計算機に作られていく事になる。

【0065】<第2の実施形態>以下、本発明の第2の実施形態について説明する。第2の実施形態では、第1の実施形態と同様に、ネットワークを介してファイルを参照複製する例について述べる。特に、ディスクレス形態の計算機を対象として、ファイル保存プログラムにお

いて利用する「ファイル保存用に割り当てられた記憶装置」として主メモリを利用することを特徴とする。

【0066】[1] 動作手順の概要

第2の実施形態では、第1の実施形態と同様の構成の元で、ファイルアクセスを行なう例について説明するが、ファイル保存プログラムによるファイルの保存先となる記憶装置として、主メモリを利用する点に特徴がある。

【0067】[2] 第2の実施形態におけるファイル保存プログラム（主メモリ）

統計情報を使ったスワップアウト処理を行なうファイル保存プログラムについて説明する。ここでは、上述の図8におけるステップS5001～ステップS5006において説明したのと同様の処理でファイル保存処理を実現するが、ファイル保存用に割り当てられた記憶装置として、主メモリの一部を割り当てている点が異なる。なお、この際の主メモリの割り当てに関しては、オペレーティングシステムの機能により実現される。例えば、RAMディスク機能を実現する手法（ステップSunOSにおいて提供されるtmpfs機能など）により割り当てる事ができる。

【0068】以上説明したように、上記各実施形態によれば、ある程度ファイルの複製を蓄積したクライアントでは既存のNFSなどとは異なり、サーバ計算機にトラブルがあったとしても、共倒れすることがなくなるという効果がある。さらに、長期的統計の蓄積からシステムの作業内容毎に設定されたファイル参照頻度を利用して破棄すべきファイルを決定することで、限られたサイズの記憶装置を、作業内容に合わせて有効利用できる。

【0069】また、上記実施形態におけるファイルサーバは、CD-ROM等のような低速大容量の記憶装置に保持したファイルを供給可能としてもよい。

【0070】なお、本発明は、複数の機器（例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど）から構成されるシステムに適用しても、一つの機器からなる装置（例えば、複写機、ファクシミリ装置など）に適用してもよい。

【0071】また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

【0072】この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0073】プログラムコードを供給するための記憶媒体としては、例えば、フロッピディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD

-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0074】また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS（オペレーティングシステム）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0075】さらに、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0076】本発明を上記記憶媒体に適用する場合、その記憶媒体には、先に説明したフローチャートに対応するプログラムコードを格納することになるが、簡単に説明すると、図12のメモリマップ例に示す各モジュールを記憶媒体に格納することになる。

【0077】すなわち、少なくとも「検索処理モジュール」「要求処理モジュール」「保存処理モジュール」および「制御処理モジュール」の各モジュールのプログラムコードを記憶媒体に格納すればよい。

【0078】ここで、検索処理モジュールは、参照が要求されたファイルを当該装置内で検索する検索処理を行うためのプログラムモジュールである。また、要求処理モジュールは、前記検索処理によって前記ファイルが検索されなかった場合、外部装置に対して通信インターフェースを介して該ファイルを要求する要求処理を行うためのプログラムモジュールである。また、保存処理モジュールは、前記要求処理による要求の応答として送信されたファイルを前記通信インターフェースを介して受信し、この受信したファイルを保存する保存処理を行うためのプログラムモジュールである。更に、制御処理モジュールは、前記受信したファイルを参照して所定の処理を遂行する制御処理を行うためのプログラムモジュールである。

【0079】

【発明の効果】以上説明したように、本発明によれば、ファイルサーバ上に存在する必要なファイルを、自装置内の記憶装置に複製することを可能とし、耐故障性に優れたシステムを提供することができる。

【0080】また、本発明によれば、ファイルのスワップアウト時において、スワップアウトすべきファイルを適切に選択することが可能となる。

【0081】

【図面の簡単な説明】

17

【図1】第1の実施形態による計算機システム及び計算機の構成を説明するブロック図である。

【図2】クライアント計算機の記憶装置104に格納される主な制御プログラムを示す図である。

【図3】ファイルサーバ側の計算機の記憶装置104に格納される主な制御プログラムを示す図である。

【図4】本実施形態におけるファイル参照複製プログラム12の制御手順を表すフローチャートである。

【図5】本実施形態におけるファイルサーバプログラムの制御手順を説明するフローチャートである。

【図6】本実施形態におけるファイル検索プログラムの制御手順を説明するフローチャートである。

【図7】ファイル参照プログラムの制御手順を説明するフローチャートである。

【図8】本実施形態におけるファイル保存プログラムの制御手順を説明するフローチャートである。

【図9】ファイルサーバ側のファイル構成の一例を示す図である。

【図10】本実施形態のファイル保存プログラム22によって構成されるクライアント側のファイル構成の一例 *20

*を示す図である。

【図11A】『プログラム開発』という作業内容を対象としたファイル参照頻度の統計情報の一例を示す図である。

【図11B】『プログラム開発』という作業内容を対象としたファイル参照頻度の統計情報の一例を示す図である。

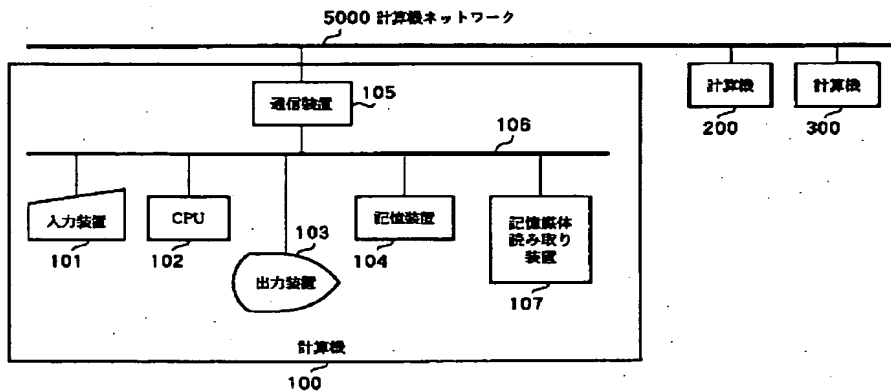
【図12】本発明に係る制御プログラムを格納する記憶媒体のメモリマップ例を示す図である。

10

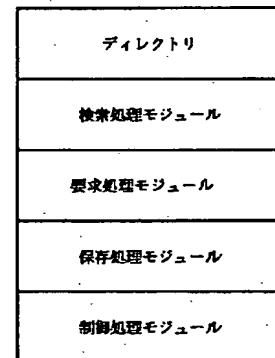
【符号の説明】

- 100、200、300 計算機
- 101 入力装置
- 102 CPU
- 103 出力装置
- 104 記憶装置
- 105 通信装置
- 106 計算機バス
- 107 記憶媒体読み取り装置
- 5000 計算機ネットワーク

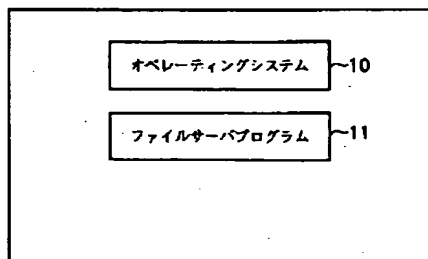
【図1】



【図12】

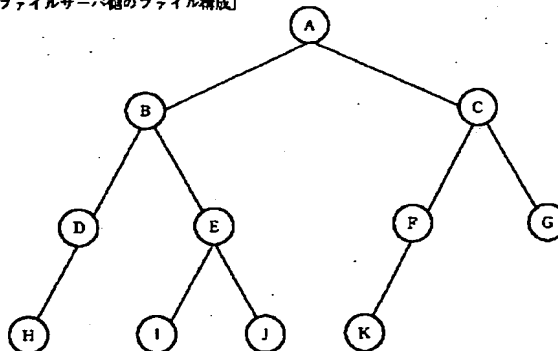


【図3】

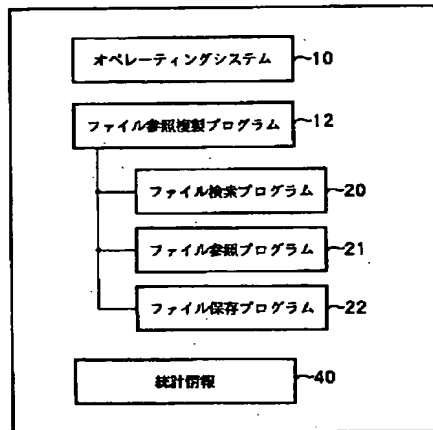


【図9】

【ファイルサーバ側のファイル構成】

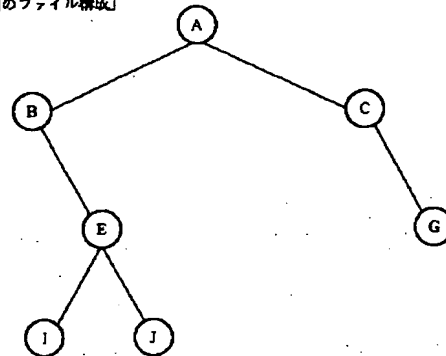


【図 2】

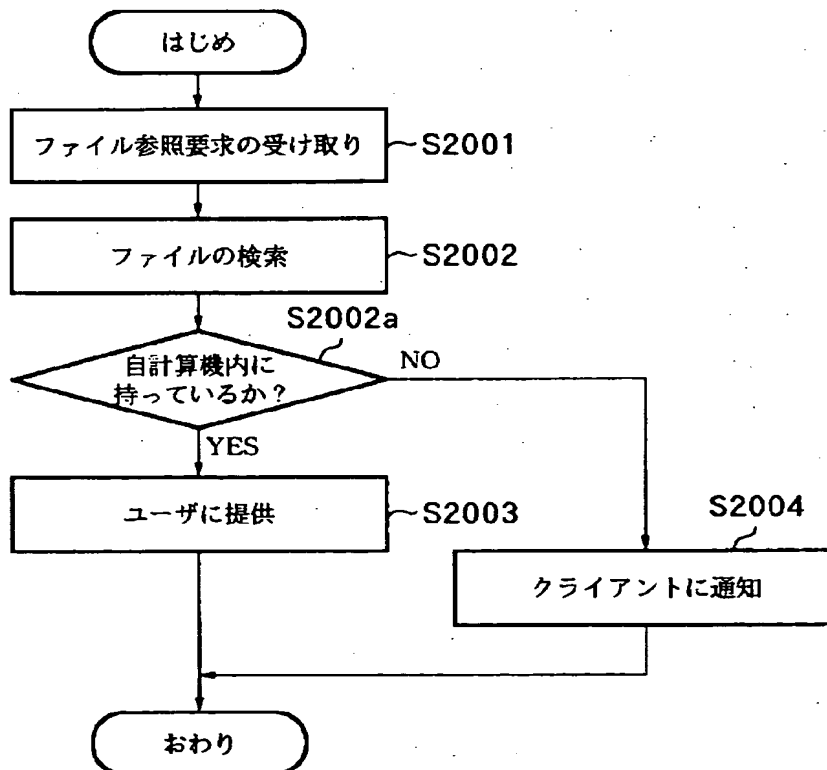


【図 10】

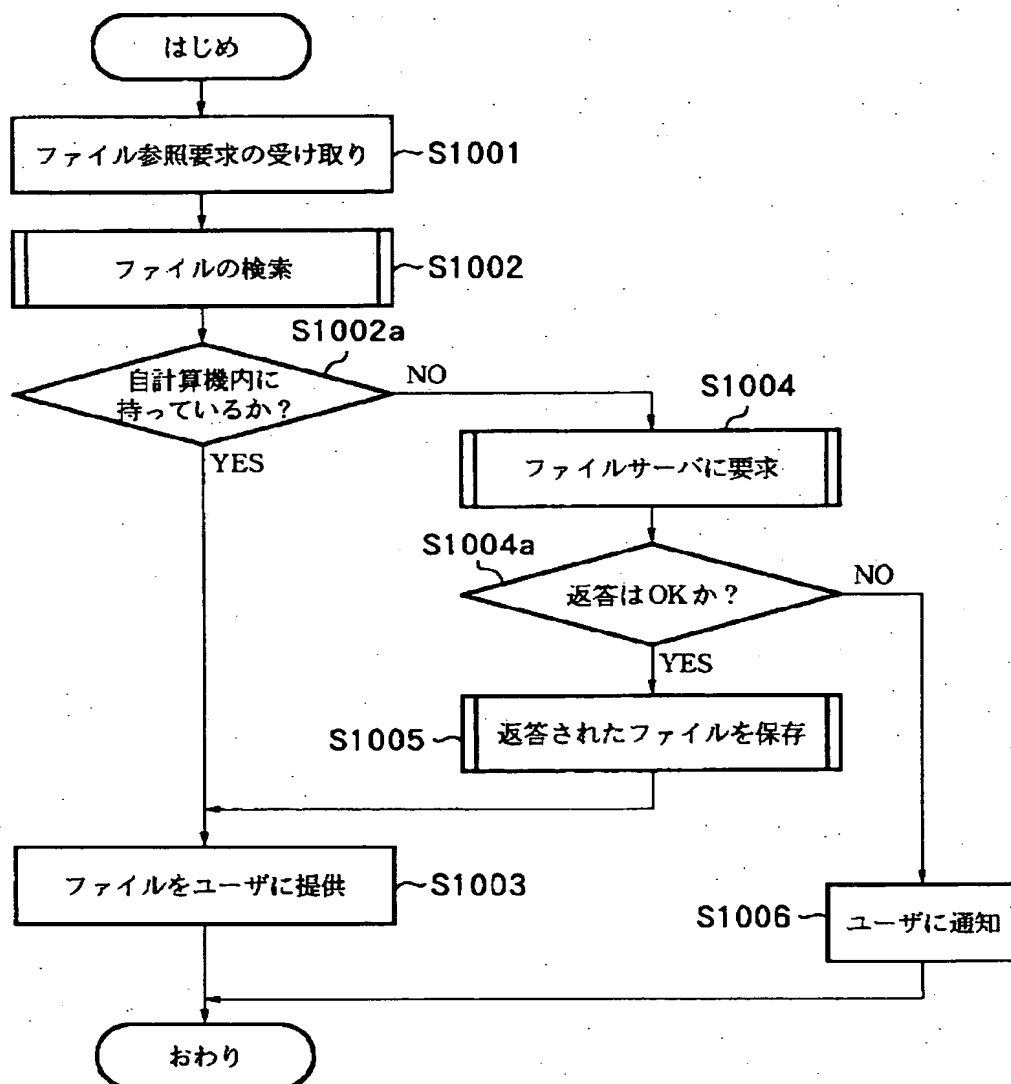
[クライアント側のファイル構成]



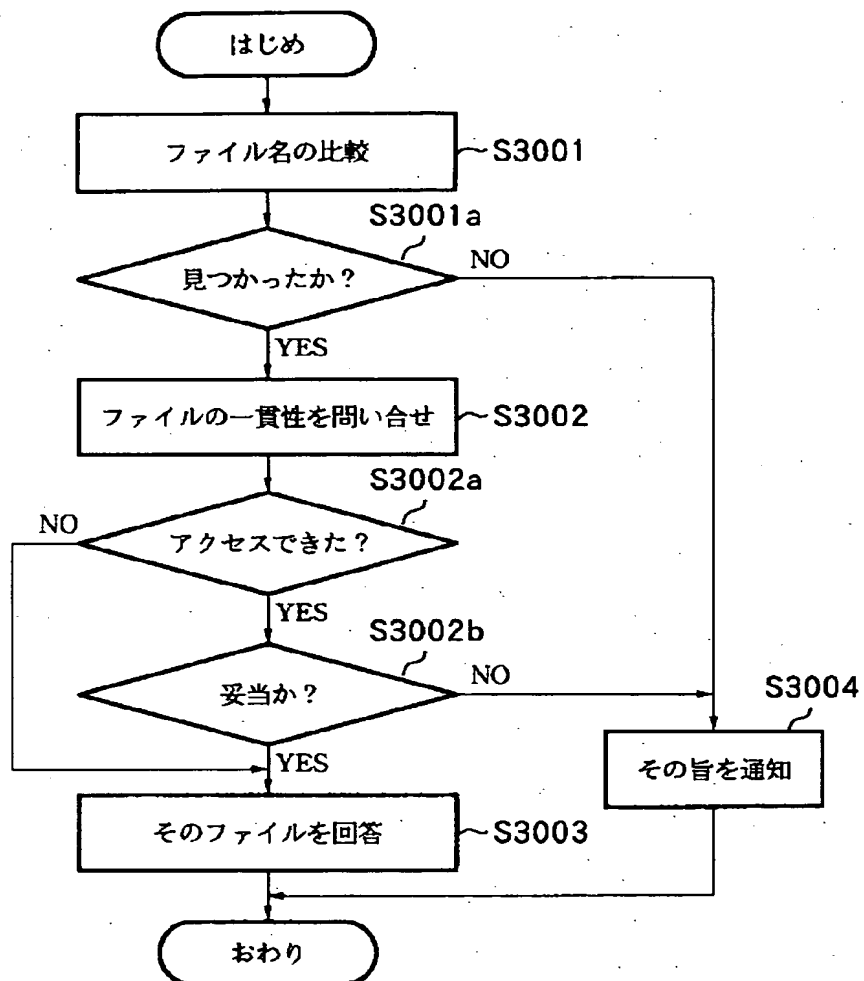
【図 5】



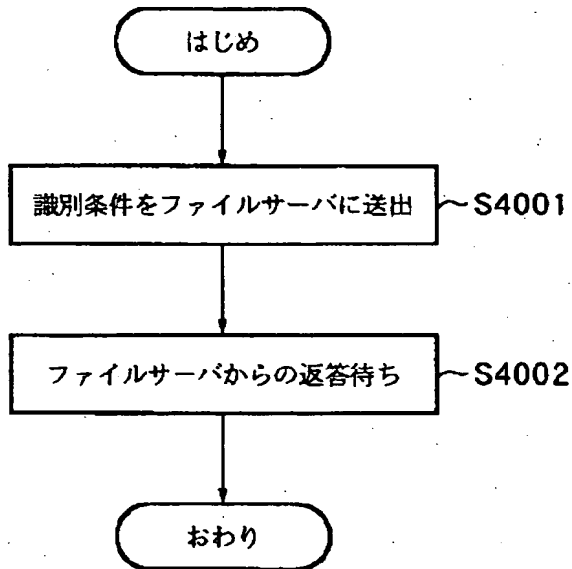
【図 4】



【図 6】



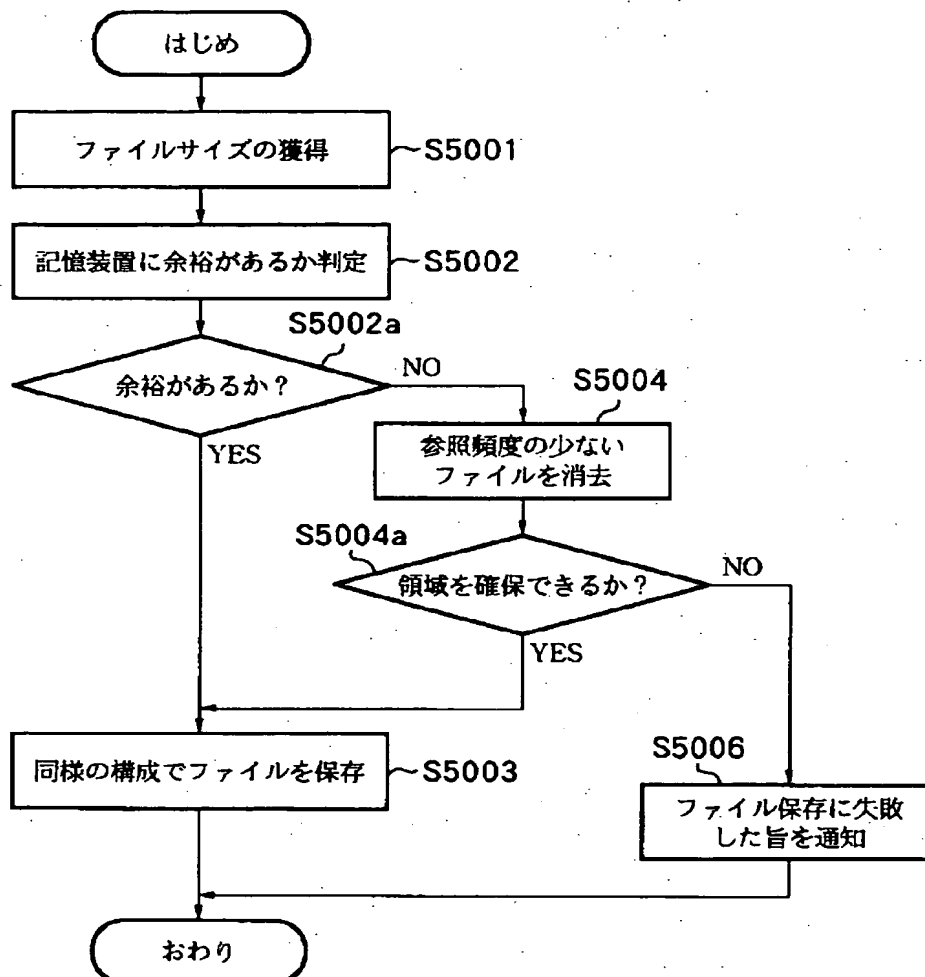
【図7】



【図11A】

ランク	頻度	コ マ ン ド	ランク	頻 度	コ マ ン ド
6	014444	/usr/bin/sh	2	000010	/usr/obj/script
	002816	/usr/lang/m		000010	/usr/obj/quota
	001661	/usr/bin/rm		000005	/usr/obj/vc
	001530	/usr/bin/hostname		000008	/usr/etc/nfsd
	001401	/usr/bin/mkdir		000008	/usr/bin/tr
	001388	/usr/lang/cpp		000008	/usr/bin/dt
	001349	/usr/lang/CC		000006	/usr/bin/chmod
	001290	/usr/bin/grep		000007	/usr/obj/more
	001266	/usr/lang/as		000007	/usr/lang/rpcgen
				000007	/usr/etc/passwd
4	000925	/usr/bin/sort	3	000006	/usr/obj/lastcomm
	000591	/usr/bin/egrep		000006	/usr/etc/cp.ftpd
	000437	/usr/bin/lis		000006	/usr/bin/ls
	000324	/usr/bin/make		000005	/usr/bin/last
	000290	/usr/bin/lid		000025	/usr/bin/stty
	000271	/usr/local/bin/gdb		000004	/usr/obj/tset
	000248	/usr/lang/c++filt		000004	/usr/etc/hiod
	000245	/usr/bin/ar		000004	/usr/bin/m4
	000235	/usr/local/bin/parch		000003	/usr/bin/cut
	000199	/usr/lang/cpp		000003	/usr/obj/ftp
3	000170	/usr/bin/ps		000003	/usr/obj/ftm
	000153	/usr/local/bin/less		000003	/usr/bin/sed
	000138	/usr/bin/ped		000003	/usr/bin/lld
	000137	/usr/obj/rub		000003	/usr/bin/crootab
	000134	/usr/bin/awk		000002	/usr/obj/printsw
	000123	/usr/bin/ranlib		000002	/usr/local/bin/gcc
	000105	/usr/bin/crash		000002	/usr/local/bin/sasica
				000002	/usr/etc/umount
				000002	/usr/etc/rcrontad
				000002	/usr/etc/lifconfig
	000090	/usr/obj/s		000002	/usr/bin/touch
	000081	/usr/bin/csh		000002	/usr/bin/file
	000067	/usr/obj/bsad			
	000054	/usr/bin/kill/xtorn			
	000047	/usr/obj/compress			
	000042	/usr/local/bin/wale			
	000037	/usr/bin/sleep			
	000034	/usr/bin/timed			
	000034	/usr/bin/cat			
	000031	/usr/bin/ps			
	000029	/usr/bin/ds			
	000027	/usr/obj/clogin			
	000019	/usr/bin/tar			
	000018	/usr/obj/rcp			
	000017	/usr/bin/date			
	000015	/usr/etc/ping			
	000015	/usr/bin/cp			
	000014	/usr/bin/kill/sasica			
	000013	/usr/bin/diff			
	000011	/usr/obj/whoami			
	000011	/usr/obj/clear			

【図 8】



【図11B】

ランク	頻度	コマン	ランク	頻度	コマン
1	000001	/usr/ach/uncompress			
	000001	/usr/bin/X11/X			
	000001	/usr/ach/whereis			
	000001	/usr/ach/tail			
	000001	/usr/ach/netstat			
	000001	/usr/ach/man			
	000001	/usr/ach/mall			
	000001	/usr/ach/e			
	000001	/usr/local/bin/gunzip			
	000001	/usr/etc/update			
	000001	/usr/etc/syslogd			
	000001	/usr/etc/portmap			
	000001	/usr/etc/nalookup			
	000001	/usr/etc/keyser			
	000001	/usr/etc/inetd			
	000001	/usr/etc/cron			
	000001	/usr/bin/wall			
	000001	/usr/bin/uname			
	000001	/usr/bin/trace			
	000001	/usr/bin/strip			
	000001	/usr/bin/rmdir			
	000001	/usr/bin/pagsize			
	000001	/usr/bin/cd			
	000001	/usr/bin/nroff			
	000001	/usr/bin/ject			
	000001	/usr/bin/dc			
	000001	/usr/bin/col			
	000001	/usr/bin/adb			
	000001	/usr/bin/X11/xinit			
	000001	/usr/bin/X11/tstun			
	000001	/usr/bin/X11/tbd node			